In re Application of LIN et al.
Serial No. 10/003,137

REMARKS

The Office action has been carefully considered. The Office action rejected claims 1-17 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,044,367 to Wolff ("Wolff"). Additionally, the Office action objected to claim 9 as not being in independent form. Applicants have amended claim 9 to be written in independent form and respectfully disagree with the rejection of claim 1-17.

By present amendment, claim 9 has been amended for clarification and not in view of the prior art. Applicants submit that the claims as filed were patentable over the prior art of record, and that the amendments herein are for purposes of clarifying the claims and/or for expediting allowance of the claims and not for reasons related to patentability. Reconsideration is respectfully requested.

Applicants thank the Examiner for the interview held (by telephone) on April 13, 2004. During the interview, the Examiner and applicants' attorney discussed the claims with respect to the prior art. The essence of applicants' position is incorporated in the remarks below.

Prior to discussing reasons why applicants believe that the claims in this application are clearly allowable in view of the teachings of the cited and applied references, a brief description of the present invention is presented.

The present invention is directed to a system and method that increases the performance of network I/O operations, particularly write and read operations directed to larger files between a network file-server and a remote client-computer. Sometimes, in a computer network environment, certain files to be written or read are quite large and may exceed certain computer systems' maximum buffer size.

As such, these large files cannot be sent or received at these limited computer systems without breaking the file into pieces. To this end, the present invention uses a client-side network redirector such that large files may be broken into smaller sections to be sent as write (or read) requests individually (by section) in a pipeline fashion to the network file server, without waiting for a response for a previously-sent section. The pipeline I/O mechanism may track the returned status of each section for which a request was made, so that success or failure of the request may be determined as a whole.

In general, the total amount of time to send an entire large file is the time spent on the first request plus the latency of the other requests that are needed. Because remote file systems frequently deal with burst traffic, most of the time a file server operates in an idle state, or a state of low workload. The present invention enables the file server to service multiple requests in parallel for the same file. Note that the above description is for example and informational purposes only, and should not be used to interpret the claims, which are discussed below.

Turning to the claims, independent claim 1 recites a method comprising, at a client-side redirector, receiving a write or read request directed to a file on a remote network server, the write or read request directed to communicating an amount of file data that exceeds a maximum buffer size allowed for communicating file data in a request to the remote server, logically separating the write or read request into a plurality of file section requests that each do not exceed the maximum buffer size, sending each of the file section requests to the remote network server, at least one file section request being sent without awaiting a status response resulting from a

previously sent file section request, tracking status information for the file section requests, and if the status information from the file section requests indicates success, returning a success indication in response to the write or read request.

The Office action rejected claim 1 as being anticipated by Wolff. More specifically, the Office action contends that Wolff teaches at a client-side redirector, receiving a write or read request directed to a file on a remote network server, the write or read request directed to communicating an amount of file data that exceeds a maximum buffer size allowed for communicating file data in a request to the remote server. Column 18, line 52 to column 19, line 5, column 48 lines 5-19, and Figs. 4C, 1A-1C, and 10G of Wolff are referenced. Further, the Office action contends that Wolff teaches logically separating the write or read request into a plurality of file section requests that each do not exceed the maximum buffer size. Column 54, lines 45-55 of Wolff are referenced. Still further, the Office action contends that Wolff teaches sending each of the file section requests to the remote network server, at least one file section request being sent without awaiting a status response resulting from a previously sent file section request. Column 65, line 66 to column 66, line 6 of Wolff are referenced. Further yet, the Office action contends that Wolff teaches tracking status information for the file section requests. Column 54, lines 45-55 of Wolff are referenced. Finally, with respect to claim 1, the Office action contends that Wolff teaches if the status information from the file section requests indicates success, returning a success indication in response to the write or read request. Column 31, lines 23-30 and Fig. 10D of Wolff are referenced. Applicants respectfully disagree.

The cited and applied reference, Wolff, is generally directed toward a system and method for alleviating the demand placed on a single server that provides files to remote computers by providing multiple servers arranged and organized in a structured and hierarchical manner. Often times, many requests for files are made simultaneously from several different remote computers. This situation taxes the resources associated with the file server system. Thus, in order to eliminate the demand placed on a single web page server, the system and method of Wolff seeks to alleviate the demand by spreading out task executions over multiple servers. As such, one server is dedicated to administrative task-assigning while the others await assignments from the administrative server. In this manner, when requests are received at the file server system, the administrative server determines which other server computers among the array of servers will respond to the request. Once assigned, the chosen server executes its assigned task.

For example, an I/O request may be received at the file server system of Wolff. The system of Wolff creates two "portions" that are based on the original I/O request: an access portion that may be likened to a header portion of a TCP/IP packet and a data portion that may be likened to the data portion of a TCP/IP packet. The access portion is then directed to a dedicated administrative server which identifies what the I/O request is and selects a particular server among the array of servers to carry out the I/O request. Once a particular server is assigned, the data portion of the I/O request is sent to the assigned server for execution.

As such, the system and method taught by Wolff falls short in several areas of teaching the recitations of the claims of the present invention.

First, the system and method taught by Wolff is directed to a server-side redirector. That is, Wolff's system deals with multiple requests that may occur simultaneously at a server system and redirects the I/O requests accordingly. Significantly, claim 1 recites a client-side redirector. That is, the method recited in claim 1 is directed toward redirecting file requests at a client-side system, particularly at a client-side system that may receive a request for a file wherein the file exceeds a maximum buffer size. Wolff does not teach a client-side redirector.

Second, the manner in which Wolff's system redirects I/O requests is based on the loads of the other servers in the server-side system. That is, the administrative server determines which other server is available to fulfill each I/O request based on each server's relative load level. A server's relative load level is not the same as a maximum buffer size as recited in claim 1. The method recited in claim 1 logically separates the file request into sections if and only if the amount of file data requested exceeds a maximum buffer size that is associated with the client computer. Wolff does not teach or even show any appreciation of comparing the size of its I/O request to anything, let alone comparing it to a maximum buffer size. Furthermore, each I/O request received by the server system in Wolff is necessarily broken into an access portion and a data portion whereas the method recited in claim 1 only logically separates the file request when its size exceeds the maximum buffer size.

Third, the manner in which the system of Wolff breaks apart the I/O request is different from the manner recited by claim 1. In Wolff, the I/O request is separated into an access portion and a data portion. The access portion is merely information about the I/O request, which, as mentioned above, is similar to a TCP/IP header. The other portion, the data portion, still contains the bulk (and subsequent large size) of the I/O request and, specifically, the portion required for processing once a server is assigned for execution by the administrative server.

Quite differently, the method recited in claim 1 logically separates the file request into a plurality of file sections wherein each one does not exceed the maximum buffer size. As stated above, Wolff does not teach or even show any appreciation of buffer size, thus Wolff cannot possibly teach separating a file request into file sections each of which is smaller than a maximum buffer size. Furthermore, the separation taught by Wolff is not a logical separation, but rather a administrative separation. That is, once separated, the access portion does not contain any actual data and the data portion does not contain any meta data. Thus, Wolff simply does not teach logically separating the write or read request into a plurality of file section requests that each do not exceed the maximum buffer size and sending each of the file section requests to the remote network server as recited in claim 1.

Fourth, claim 1 recites that at least one file section request being sent without awaiting a status response resulting from a previously sent file section request. That is, when the logical separation of the original file request has been carried out, each file section may be sent one after another without awaiting a

response from the resource in the client-computer responding to the file request. In direct contrast, the system taught by Wolff must necessarily wait for a response to sending the first portion, the access portion, because the administrative server must first assign another server to handle the data portion of the I/O request before the data portion can be sent to the assigned server. As before, Wolff fails to teach yet another recitation of claim 1.

Finally, with respect to claim 1, Wolff does not teach tracking status information for the file section requests, and if the status information from the file section requests indicates success, returning a success indication in response to the write or read request. The cited and applied section of Wolff teaches a status indication of whether or not each server is mounted and/or available for assignment by the administrative server. Status indication of the drive or server itself is not the same as status indication of a file request directed to the drive or server.

Applicants submit that claim 1 is patentable over the prior art of record for at least the foregoing reasons.

Applicants respectfully submit that dependent claims 2-8, by similar analysis, are also allowable. Each of these claims depends either directly or indirectly from claim 1 and consequently includes the recitations of independent claim 1. As discussed above, Wolff fails to disclose the recitations of claim 1 and, therefore, these claims are also allowable over the prior art of record. In addition to the recitations of claim 1 noted above, each of these dependent claims includes additional patentable elements.

For example, claim 8 recites that the method of claim 1 further comprises determining that the write or read request exceeds the maximum buffer size allowed. As discussed above, Wolff does not teach or even show any appreciation of a maximum buffer size, thus Wolff cannot possibly teach comparing read or write requests to a maximum buffer size. Applicants submit that claim 8 is allowable over the prior art of record for at least this additional reason.

Turning to the next independent claim, amended claim 9 recites a computer-readable medium having computer-executable instructions for receiving, at a client-side redirector, a write or read request directed to a file on a remote network server, the write or read request directed to communicating an amount of file data that exceeds a maximum buffer size allowed for communicating file data in a request to the remote server, logically separating the write or read request into a plurality of file section requests that each do not exceed the maximum buffer size, sending each of the file section requests to the remote network server, at least one file section request being sent without awaiting a status response resulting from a previously sent file section request, tracking status information for the file section requests, and if the status information from the file section requests indicates success, returning a success indication in response to the write or read request.

In essence, claim 9 has been amended to include the recitations of claim 1 as requested by the Office action. As such, the computer-readable medium of claim 9 recites limitations similar to the limitation recited by claim 1. Thus, applicants submit that claim 9 is patentable over the prior art of record for at least the reasons set forth above with respect to claim 1.

Turning to the next independent claim, claim 10 recites a client-side system, comprising an application program that issues an I/O request corresponding to a file on a file server, the I/O request corresponding to an amount of file data that exceeds an allowed amount that can be exchanged with the file server in a single request, and a network redirector having an associated pipeline I/O mechanism, configured to: 1) receive information corresponding to the I/O request; 2) send a plurality of sectioned I/O requests to the network server to satisfy the I/O request received at the application, at least one of the requests sent without awaiting status information from the server for a previously sent request, and each sectioned I/O request corresponding to file data that does not exceed the allowed amount; 3) track status information for each of the sectioned I/O requests; and 4) determine a status to return to the application program based on the tracked status information.

Significantly, Wolff fails to teach several of the recitations found in claim 10. For example, the system of Wolff is directed to a server-side redirector as opposed to a client-side redirector as recited in claim 10. Further, Wolff fails to teach an application program that issues an I/O request corresponding to a file on a file server, the I/O request corresponding to an amount of file data that exceeds an allowed amount that can be exchanged with the file server in a single request. That is, Wolff does not teach or even show any appreciation of handling an I/O request that exceeds an allowed amount that can be exchanged with the file server in a single request, *i.e.* a maximum buffer size. Further yet, Wolff fails to teach sending a plurality of sectioned I/O requests to the network server to satisfy the I/O request received at the application, at least one of the requests sent without

awaiting status information from the server for a previously sent request. Wolff only sends a single section (the data section) to fulfill the I/O request (after a server has been assigned) and, thus, cannot send another section because no other sections exist. Finally, with respect to claim 10, Wolff fails to teach track status information for each of the sectioned I/O requests as discussed above.

Applicants submit that claim 10 is allowable over the prior art of record for at least these reasons.

Applicants respectfully submit that dependent claims 11-16, by similar analysis, are also allowable. Each of these claims depends either directly or indirectly from claim 10 and consequently includes the recitations of independent claim 10. As discussed above, Wolff fails to disclose the recitations of claim 10 and, therefore, these claims are also allowable over the prior art of record. In addition to the recitations of claim 10 noted above, each of these dependent claims includes additional patentable elements.

Turning to the last independent claim, claim 17 recites a method comprising at a client-side redirector, receiving a write request to write file data to a file on a remote network server, the write request indicating an amount of data to be written that exceeds a maximum size allowed per request by the remote server logically separating the write request into a plurality of partial write requests that each do not exceed the maximum buffer size, allocating a data structure containing an entry for each file section request, sending each of the file section requests to the remote network server, at least one request being sent without awaiting a status response that results from a previously sent request, evaluating responses from the file

system, and for each successful response that corresponds to a partial write request, updating the array at a location therein that corresponds to that file section request, and returning a success indication when the array indicates that each of the partial write requests was successful.

Applicants submit that claim 17 is allowable for at least the reasons discussed above with respect to claim 1 and claim 10. Several of the recitations of claim 17, including a client-side redirector and logically separating the write request into a plurality of partial write requests that each do not exceed the maximum buffer size, are simply not taught by Wolff as discussed above regarding claim 1 and claim 8. As such, applicants submit that claim 17 is patentable over the prior art of record.

For at least these additional reasons, applicants submit that all the claims are patentable over the prior art of record. Reconsideration and withdrawal of the rejections in the Office Action is respectfully requested and early allowance of this application is earnestly solicited.

In re Application of LIN et al.
Serial No. 10/003,137

CONCLUSION

In view of the foregoing remarks, it is respectfully submitted that claims 1-17 are patentable over the prior art of record, and that the application is good and proper form for allowance. A favorable action on the part of the Examiner is earnestly solicited.

If in the opinion of the Examiner a telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney at (425) 836-3030.

Respectfully submitted,

Albert S. Michalik, Reg. No. 37,395
Attorney for Applicants
Law Offices of Albert S. Michalik, PLLC
704 - 228th Avenue NE, Suite 193
Sammamish, WA 98074
(425) 836-3030
(425) 836-8957 (facsimile)

In re Application of LIN et al.
Serial No. 10/003,137

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this Amendment, along with Transmittal, Petition for Extension of Time and Facsimile cover sheet, are being transmitted by facsimile to the United States Patent and Trademark Office in accordance with 37 C.F.R. 1.6(d) on the date shown below:

Date: July 6, 2004

Albert S. Michalik

2960 Amendment